Math 229. 2020-Sep-11

Lab 3. More Graphing.

Recall just a number * vector

vector .* vector

Demo Example 1 ✓
Example 3 ✓

10:30 - 11:30 You can stay, or log off, or log off & log back in when you have Questions (about Lab 1, 2, or 3).
We're doing Lab 3

Also: office hours today 2pm

zoom.us/my/mattsunderland

Optional: Making a table of x,y values

```
t = 0:10;
y = 2 + 10*t - 1/2*t.^2;
[t;y]'
```

this makes a table

eg,

t	y
0	2
1	7
2	2
3	-13
⋮	⋮

Lab 2 Question 10

just "translate" into MATLAB

sin(x)cos(x) ⟶ $\boxed{\texttt{sin(x).*cos(x)}}$

3 Identifying an appropriate domain

MATLAB can't think for us, but it can help us "think easier." Case in point, we're often interested in plotting a function on only part of its domain. This may be where a function makes sense for a physical problem such as non-negative time, or non-negative area. Or, it may be where we can identify one cycle of a periodic function. We can analyze the function to do this, or we can use MATLAB to explore some graphs and from these decide upon an appropriate domain.

Example 1:
A baseball is thrown from center field towards home plate. Its height, y, varies as a function of time, t. It is known from high-school physics that a model for this is given by the projectile motion formula

$$y = y_0 + v_0 t - \frac{1}{2} a t^2,$$

so for this example, stop when $y(t) = 0$ again

where y_0 and v_0 are the initial height and upward velocity, and a is the constant of acceleration. In metric units of meters and seconds, we will assume $a = 10$ instead of the more accurate 9.8 m/s^2.

Suppose a ball is thrown with an initial velocity of $v_0 = 10$ meters per second upward from an initial height of $y_0 = 2$ meters. On what realistic domain will $y(t) \geq 0$?

Mathematically, $y(t)$ is a parabola which opens downward as the t^2 coefficient is negative. We could use formulas to find a parabola's zeroes, but using MATLAB we can have fun exploring the graph just as easily. First, we make an initial plot of the data over the interval $[0, 10]$. Why that? Well time must be non-negative so $t \geq 0$ is natural. As for 10, since this is supposed to model a ball being thrown, 10 seconds should be long enough, if the model is realistic.

```
>> t = linspace(0,10);
>> y = 2 + 10*t - (1/2)*10*t.^2;
>> plot(t,y)
```

If you make this graph, you'll see that 10 seconds was too long. (Why). Let's replot using an interval of $[0, 2]$.

```
>> t = linspace(0,2);
>> plot(t,2 + 10*t - (1/2)*10*t.^2)
```

(We combined the definition of `y` directly into the `plot` function.)

Now our interval is too small.

Exercise 2:
Replot the ball problem until you can find a good estimate for the value of b so that $y(t) \geq 0$ on the interval $[0, b]$, and is negative for $t > b$. What is the value of b that you found?
(4) Answer: ______________________________

Exercise 3:
Find the suggested values by plotting the graphs until you can figure out the answer.

a. Repeating the above example, find the period of the function

$$f(x) = 120\sin(120\pi x).$$

(This is a bit tricky! Keep plotting until you clearly get one period shown in the viewing window.)
(5) Answer: (the answer is a number)

b. When does $f(x) = 25 - x - \sin(x)$ cross the x-axis? (Guess an answer to within 1 decimal point.)
(6) Answer: ______________________________

4 Plotting functions with vertical asymptotes

Plotting functions with vertical asymptotes can cause troubles. For instance, suppose we want to plot the function $f(x) = 1/\sin(x)$ over the interval $[0, \pi]$. A first attempt may look like this:

```
>> x = linspace(0,pi);
>> plot(x, 1./sin(x))
```

If you make this graph, you'll find that it doesn't look like what you would expect. In this case the function has a vertical asymptote at both endpoints 0 and π. This means that when the x values are close to the endpoints, the corresponding y values get to be very large. This is why the y-axis is labeled using scientific notation indicating 15 zeroes after the numbers. This makes this graph worthless for answering questions about $f(x)$. The remedy is to replot, only this time avoiding the asymptotes:

```
>> delta = 0.1
>> x = linspace(delta, pi-delta)
>> plot(x, 1./sin(x))
```

This is much better, allowing us to see the shape of the graph. We used a variable `delta` so that if we wanted to, it is easy to make changes.

Exercise 4:
From its graph, estimate the minimum value of the function

$$f(x) = \frac{5}{\cos(x)} + \frac{8}{\sin(x)}$$

over the interval $(0, \pi/2)$.

Just plotting with x values given by

domain

Example 3:
Plot the functions $f(x) = 4\cos x$ and $g(x) = \cos 4x$ together over the interval $0 \le x \le 2\pi$.
Solution: We first create pairs of arrays as if we were to graph each function separately.

```
>> x = linspace(0,2*pi);
>> y1=4*cos(x);                  % this is the first function
>> y2=cos(4*x);                  % the second function needs a different name
>> plot(x,y1,x,y2), grid         % plot(x,y) is one graph. This makes two
```

MATLAB creates a frame in which all function values fit, and then plots the graph of each function. Different colors are used for each function.

If we prefer, `plot(x,y1,x,y2)` can be replaced with:

```
>> plot(x,y1)
>> hold on            % next plot will be on top of the current one
>> plot(x,y2)
>> hold off           % turn off the hold. Next graph opens a new window
```

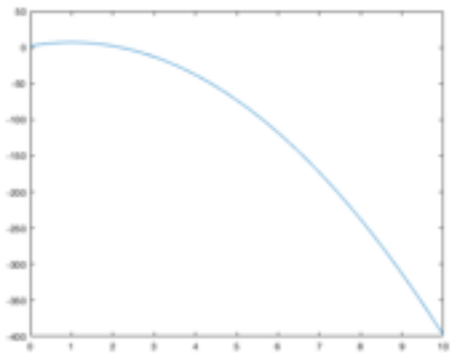
The `hold on` command instructs MATLAB not to create a new plot window with subsequent `plot` calls. By entering `hold off` at the end of the example, we allow new plot windows to be created.

Exercise 6:
Consider the functions of the previous example, $y_1 = 4\cos(x)$ and $y_2 = \cos(4x)$, make the graphs and use them to answer the following:

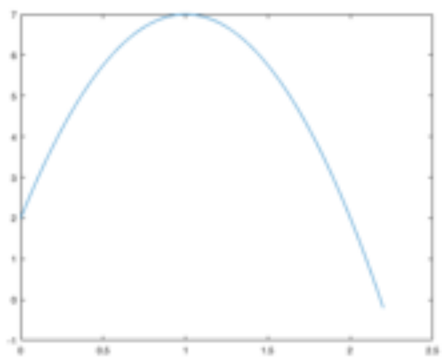
a. Which function is oscillating more rapidly?
   (11) Circle one:
   1. $\cos(4x)$
   2. $4\cos(x)$

b. Which function has the larger amplitude?
   (12) Circle one:
   1. $\cos(4x)$
   2. $4\cos(x)$

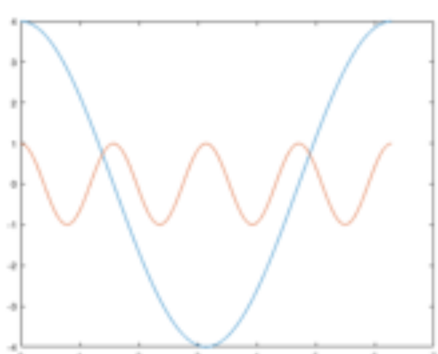
```
t = linspace(0,10);
y = 2 + 10*t - 1/2*10*t.^2;
plot(t,y)
```



```
t = linspace(0,2.2);
y = 2 + 10*t - 1/2*10*t.^2;
plot(t,y)
```



```
x = linspace(0,2*pi);
f = 4*cos(x);
g = cos(4*x);
plot(x,f,x,g)
```

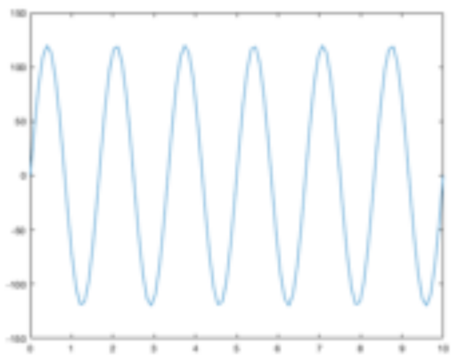


```
t = 0:10;
y = 2 + 10*t - 1/2*10*t.^2;
[t;y]'

t = 2:0.1:3;
y = 2 + 10*t - 1/2*10*t.^2;
[t;y]'

t = 2.1:0.01:2.2;
y = 2 + 10*t - 1/2*10*t.^2;
[t;y]'
```

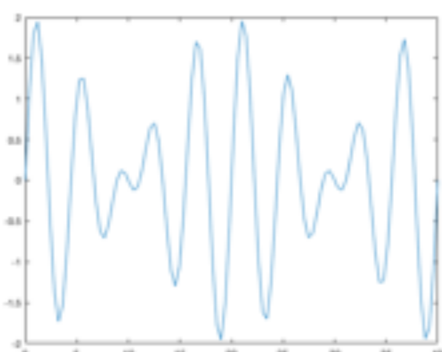
```
ans =

     0     2
     1     7
     2     2
     3   -13
     4   -38
     5   -73
     6  -118
     7  -173
     8  -238
     9  -313
    10  -398


ans =

    2.0000    2.0000
    2.1000    0.9500
    2.2000   -0.2000
    2.3000   -1.4500
    2.4000   -2.8000
    2.5000   -4.2500
    2.6000   -5.8000
    2.7000   -7.4500
    2.8000   -9.2000
    2.9000  -11.0500
    3.0000  -13.0000


ans =

    2.1000    0.9500
    2.1100    0.8395
    2.1200    0.7280
    2.1300    0.6155
    2.1400    0.5020
    2.1500    0.3875
    2.1600    0.2720
    2.1700    0.1555
    2.1800    0.0380
```

```
    2.1900   -0.0805
    2.2000   -0.2000
```

```
x = linspace(0,10);
f = 120*sin(120*pi*x);
plot(x,f)
```



```
x = linspace(0,40);
y = sin(pi/2*x) + sin(2/5*pi*x);
plot(x,y)
```



Published with MATLAB® R2020a